

☆ Build the Subsequences

A subsequence of string, s, is obtained by deleting one or more characters from s. For example, the set of subsequences for string s = "abc" would be $\{"a", "ab", "ac", "abc", "b", "bc", "c", ""\}$ (recall that the empty string is a subsequence of all strings).

Complete the *buildSubsequences* function in your editor. It has 1 parameter: a string, s. It must return an array of strings containing all possible subsequences of s in *alphabetical* order. *Do not* include the empty string in your returned array of subsequences.

Input Format

The locked stub code in your editor reads a single string, s, from stdin and passes it to your function.

Constraints

- 1 < |s| < 16
- s is a string of unique lowercase letters (a-z).

Output Format

Output to stdout is handled by the locked stub code in your editor, which prints each element of the returned array on a new line.

ba

Sample Output 1 a b ba

Sample Input 2

abc

Sample Output 2

```
a
ab
abc
ac
b
bc
c
```

Explanation

Sample Case 1

s = ba

There are 3 subsequences b, a, ba

Sample Case 2

s = abc

There are 7 subsequences a, ab, abc, ac, b, bc, c

YOUR ANSWER

We recommend you take a quick tour of our editor before you proceed. The timer will pause up to 90 seconds for the tour.

Start tour

```
Original code

1 ▶ import ↔;
6
7 public class Solution {
```

×

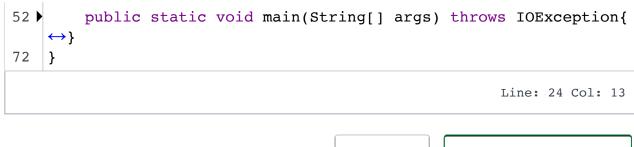
```
9 - /*
10
     * Complete the function below.
     */
11
12
13 ▼
        static String[] buildSubsequences(String s) {
14 <del>-</del>
             if (s == null | s.length() == 0) {
                 return null;
15
16
             }
17
18
             LinkedHashSet<String> set = new LinkedHashSet<String>
    ();
19
             dfs(set, 0, new StringBuilder(), s);
20
21
             String[] res = new String[set.size()];
             int i = 0;
22
             for (String sub : set) {
23 ₩
                 res[i] = sub;
24
25
                 i++;
26
             }
27
```



Fall 2016 - CMU OCI

© 25m: 47s to test end

```
32
                }
       33
                static void dfs(LinkedHashSet<String> set, int start,
       34 ▼
           StringBuilder tmp, String s) {
                    if (start == s.length()) {
       35 ₩
       36
                        String sub = tmp.toString();
1
                         if (sub.length() > 0) {
       37 ₹
                            set.add(sub);
       38
2
       39
                         }
       40
                        return;
3
       41
                    }
       42
       43
                    dfs(set, start + 1, tmp, s);
       44
                    tmp.append(s.charAt(start));
       45
                    dfs(set, start + 1, tmp, s);
                    tmp.deleteCharAt(tmp.length() - 1);
       46
       47
       48
                    return;
       49
                }
       50
       51
```



Test against custom input

Run Code Submit code & Continue

(You can submit any number of times)

▲ Download sample test cases The input/output files have Unix line endings. Do not use Notepad to edit them on windows.





1

2

3

Testcase 1: Success

Your Output

b ba

а

Expected Output

а b



Your Output

а ab abc ac

b bc С **Expected Output** а ab abc ac b bc С Testcase 3: Success **Your Output** Output hidden **O** 25m : 47s Fall 2016 - CMU OCI to test end Output hidden **Testcase 5: Success Your Output** 1 Output hidden 2 **Testcase 6: Success** 3 **Your Output** Output hidden **Testcase 7: Success Your Output** Output hidden **Testcase 8:** Success **Your Output**

Output hidden

Privacy Policy Terms of Service About



Fall 2016 - CMU OCI



O 25m: 47s

to test end





1

2

3

